

AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions, and listings, of claims in the application:

1. 1. (Currently Amended) A database system comprising:
processors;
a storage subsystem having plural storage modules to store respective portions of a view containing results of a cube-based operation on at least one base table, the view containing a first result set for a group-by on a first grouping set, and a second result set for a group-by on a second grouping set, wherein the second grouping set has less attributes than the first grouping set; and
database software executable on the processors and having plural access modules that provide parallel access of data in the plural storage modules, wherein the database software executable on the processors is configured to:
distribute the portions of the view across the plural storage modules based on hash values computed by hashing plural attributes specified by a function defining the view;
~~a controller~~, in response to a change to the at least one base table, to:
update the first result set by computing a change to the first result set based on [[a]]the change in the at least one base table;[[and]]
distribute rows of the change to the first result set across the plural storage modules based on hash values computed by hashing the attributes of the second grouping set and at least one other attribute that is part of the first grouping set but not the second grouping set,
wherein the at least one other attribute is assigned a predefined value to perform the hashing of the attributes of the second grouping set and the at least one other attribute; and
update the second result set by computing a change to the second result set based on the distributed rows of the change to the first result set.
2. 2. (Original) The database system of claim 1, wherein the first grouping set has a first number of grouping attributes, and the second grouping set has a second number of grouping attributes, the first number being greater than the second number.
1. 3. (Cancelled)

1 4. (Original) The database system of claim 1, wherein the view contains results of a group-
2 by cube operation.

1 5. (Original) The database system of claim 1, wherein the view contains results of a group-
2 by partial cube operation.

1 6.-7. (Cancelled)

1 8. (Currently Amended) The database system of claim [[7]]1, wherein the function is a cube
2 function, and the view contains results of a cube operation specified by [[a]]the cube function on
3 the plural eolumns_attributes, and wherein the at least another eolumn being attribute is one of the
4 plural eolumns_attributes of the cube function that is not in the second grouping set.

1 9. (Currently Amended) The database system of claim [[7]]1, wherein the view is
2 distributed across the access modules such that plural portions of the view reside in respective
3 storage modules, and wherein the rows [[in]]of the change to the first result set are distributed
4 across the access-storage modules according to the hash-hashing of the attributes of the second
5 grouping set and the at least one other attribute to enable:

6 each access module to locally perform a merge and aggregate operation on the rows of
7 change to the first result set to produce rows of update the second result set; and

8 each access module to locally merge the rows of the change to the second result set into a
9 respective portion of the view without having to first redistribute the rows of the change to the
10 second result set.

1 10. (Currently Amended) The database system of claim 1, wherein the controller-database
2 software is adaptedconfigured to further:

3 receive a query specifying a group-by operation; and

4 determine whether an answer for the query specifying the group-by operation can be
5 satisfied from the view.

1 11. (Currently Amended) The database system of claim 10, wherein the query specifies a
2 group-by operation on grouping sets S, and the view contains result sets for grouping sets C,
3 the controller database software being configured to determine whether S is a subset of C
4 to determine whether the answer for the query can be satisfied from the view.

1 12. (Currently Amended) The database system of claim 11, wherein the controller database
2 software is adapted configured to modify a WHERE clause of the query in response to
3 determining that S is a subset of C.

1 13. (Currently Amended) A method for use in a database system, comprising:
2 storing a view containing results of a cube-based operation on at least one base table, the
3 view containing result sets for group-bys on respective grouping sets, wherein portions of the
4 view are distributed across plural storage modules of a storage subsystem, wherein the result sets
5 include a first result set and a second result set, wherein the first result set is for a group-by on a
6 first of the grouping sets, and the second result set is for a group-by on a second of the grouping
7 sets, and wherein the second grouping set has less attributes than the first grouping set;
8 accessing data in the plural storage modules in parallel by corresponding plural access
9 modules in database software executing on processors;
10 distributing, by the processors, the portions of the view across the plural storage modules
11 based on hash values computed by hashing plural attributes specified by a function defining the
12 view;
13 updating[[a]], by the processors, the first result set by computing a change to the first
14 result set based on a change in the at least one base table;[[and]]
15 distributing, by the processors, rows of the change to the first result set across the plural
16 storage modules based on hash values computed by hashing the attributes of the second grouping
17 set and at least one other attribute that is part of the first grouping set but not the second grouping
18 set, wherein the at least one other attribute is assigned a predefined value to perform the hashing
19 of the attributes of the second grouping set and the at least one other attribute; and
20 updating, by the processors, the[[a]] second result set by computing a change to the
21 second result set based on the distributed rows of the change to the first result set.

1 14. (Cancelled)

1 15. (Original) The method of claim 13, further comprising updating a third result set by
2 computing a change to the third result set based on the change to the second result set.

1 16. (Original) The method of claim 15, further comprising updating a fourth result set by
2 computing a change to the fourth result set based on the change to the third result set.

1 17.-18. (Cancelled)

1 19. (Currently Amended) The method of claim [[18]]13, wherein the first result set
2 corresponds to [[a]]the group-by on [[a]]the first grouping set having N columnsattributes, and
3 the second result set corresponds to [[a]]the group-by on [[a]]the second grouping set having N-1
4 columnsattributes, and wherein the N-1 attributes in the second group set are all part of the first
5 grouping set, and wherein distributing the first result set to compute the second result set
6 comprises distributing the first result set based on a hash of the N columns, with the column in
7 the first grouping set not present in the second grouping set being assigned a predefined value.

1 20. (Currently Amended) The method of claim 19, wherein the result sets of the view further
2 comprise a third result set for a group-by on a third of the grouping sets, wherein the third
3 grouping set has less attributes than the second grouping set, the method further comprising:
4 distributing rows of the change to the second result set across the plural storage modules
5 based on hash values computed by hashing the attributes of the third grouping set and at least
6 two other attributes that are part of the first grouping set but not the third grouping set, wherein
7 the at least two other attributes are assigned the predefined value to perform the hashing of the
8 attributes of the third grouping set and the at least two other attributes;
9 updating [[a]]the third result set by computing a change to the third result set based on the
10 distributed rows of the change to the second result set, wherein the third result set corresponds to
11 a group by on a third grouping set having N-2 columns,
12 wherein distributing the second result set across the access modules to compute the third
13 result set is based on a hash of the N columns, with the columns in the first grouping set not
14 appearing in the third grouping set each being assigned to the predefined value.

1 21. (Currently Amended) The method of claim 20, wherein the function is a cube function,
2 and wherein storing the view comprises storing [[a]]the view for a cube operation based on
3 [[a]]the cube function of the N columnsattributes.

1 22. (Currently Amended) An article comprising at least one storage medium containing
2 instructions that when executed cause a database system to:

3 store a view containing results of a cube-based operation on at least one base table, the
4 view containing result sets for group-bys on respective grouping sets, wherein portions of the
5 view are distributed across plural storage modules of a storage subsystem, wherein the result sets
6 include a first result set and a second result set, wherein the first result set is for a group-by on a
7 first of the grouping sets, and the second result set is for a group-by on a second of the grouping
8 sets, and wherein the second grouping set has less attributes than the first grouping set;

9 access data in the plural storage modules in parallel by corresponding plural access
10 modules in database software in the database system;

11 distribute the portions of the view across the plural storage modules based on hash values
12 computed by hashing plural attributes specified by a function defining the view;

13 update [[a]]the first result set by computing a change to the first result set based on a
14 change in the at least one base table; and

15 distribute rows of the change to the first result set across the plural storage modules based
16 on hash values computed by hashing the attributes of the second grouping set and at least one
17 other attribute that is part of the first grouping set but not the second grouping set, wherein the at
18 least one other attribute is assigned a predefined value to perform the hashing of the attributes of
19 the second grouping set and the at least one other attribute; and

20 update [[a]]the second result set by computing a change to the second result set based on
21 the distributed rows of the change to the first result set.

1 23. (Cancelled)

1 24. (Original) The article of claim 22, wherein the instructions when executed cause the
2 database system to further update a third result set by computing a change to the third result set
3 based on the change to the second result set.

1 25.-26. (Cancelled)

1 27. (Currently Amended) The article of claim [[26]]13, wherein the first result set
2 corresponds to [[a]]the group-by on [[a]]the first grouping set having N columnsattributes, and
3 the second result set corresponds to [[a]]the group-by on [[a]]the second grouping set having N-1
4 columnsattributes, and wherein the N-1 attributes in the second group set are all part of the first
5 grouping set, and wherein distributing the first result set to compute the second result set
6 comprises distributing the first result set based on a hash of the N columns, with the column in
7 the first grouping set not present in the second grouping set being assigned a predefined value.

1 28. (Currently Amended) The article of claim 27, wherein the function is a cube function,
2 and wherein storing the view comprises storing [[a]]the view for a cube operation based on
3 [[a]]the cube function of the N columnsattributes.

4
5 29. (New) The database system of claim 1, wherein the view further contains a third result set
6 for a group-by on a third grouping set having less attributes than the second grouping set, and
7 wherein the database software is configured to further:

8 distribute rows of the change to the second result set across the plural storage modules
9 based on hash values computed by hashing the attributes of the third grouping set and at least
10 two other attributes that are part of the first grouping set but not the third grouping set, wherein
11 the at least two other attributes are assigned the predefined value to perform the hashing of the
12 attributes of the third grouping set and the at least two other attributes; and

13 update the third result set by computing a change to the third result set based on the
14 distributed rows of the change to the second result set.

1 30. (New) The database system of claim 29, wherein all of the attributes of the second
2 grouping set are part of the first grouping set, and all of the attributes of the third grouping set are
3 part of the second grouping set.

1 31. (New) The article of claim 22, wherein the result sets of the view further include a third
2 result set for a group-by on a third of the grouping sets having less attributes than the second
3 grouping set, and wherein the instructions when executed cause the database system to further:
4 distribute rows of the change to the second result set across the plural storage modules
5 based on hash values computed by hashing the attributes of the third grouping set and at least
6 two other attributes that are part of the first grouping set but not the third grouping set, wherein
7 the at least two other attributes are assigned the predefined value to perform the hashing of the
8 attributes of the third grouping set and the at least two other attributes; and
9 update the third result set by computing a change to the third result set based on the
10 distributed rows of the change to the second result set.